What order does the DIR command arrange files if no sort order is specified?

devblogs.microsoft.com/oldnewthing/20140304-00

March 4, 2014

Raymond Chen

If you don’t specify a sort order, then the `DIR` command lists the files in the order that the files are returned by the `FindFirstFile` function.

Um, okay, but that just pushes the question to the next level: What order does `FindFirstFile` return files?

The order in which `FindFirstFile` returns files in unspecified. It is left to the file system driver to return the files in whatever order it finds most convenient.

Now we’re digging into implementation details.

For example, the classic FAT file system simply returns the names in the order they appear on disk, and when a file is created, it is merely assigned the first available slot in the directory. Slots become available when files are deleted, and if no slots are available, then a new slot is created at the end.

Modern FAT (is that an oxymoron?) with long file names is more complicated because it needs to find a sequence of contiguous entries large enough to hold the name of the file.

There used to be (maybe there still are) some low-level disk management utilities that would go in and manually reorder your directory entries.

The NTFS file system internally maintains directory entries in a B-tree structure, which means that the most convenient way of enumerating the directory contents is in B-tree order, which if you cover one eye and promise not to focus too closely looks *approximately* alphabetical for US-English. (It’s not very alphabetical for most other languages, and it falls apart once you add characters with diacritics or anything outside of the Latin alphabet, and that includes spaces and digits!)

The ISO 9660 file system (used by CD-ROMs) requires that directory entries be lexicographical sorted by ASCII code point. Pretty much everybody has abandoned the base ISO 9660 file system and uses one of its many extensions, such as Joliet or UDF, so you have that additional wrinkle to deal with.

If you are talking to a network file system, then the file system on the other end of the network cable could be anything at all, so who knows what its rules are (if it even has rules).

When people ask this question, it's usually in the context of a media-playing device which plays media from a CD-ROM or USB thumb drive in the raw native file order. But they don't ask this question right out; they ask some side question that they think will solve their problem, but they don't come out and say what their problem is.

So let's *solve the problem in context*: If the storage medium is a CD-ROM or an NTFS-formatted USB thumb drive, then the files will be enumerated in sort-of-alphabetical order, so you can give your files names like `000 First track.mp3`, `001 Next track.mp3`, and so on.

If the storage medium is a FAT-formatted USB thumb drive, then the files will be enumerated in a complex order based on the order in which files are created and deleted and the lengths of their names. But the easy way out is simply to remove all the files from a directory then move file files into the directory in the order you want them enumerated. That way, the first available slot is the one at the end of the directory, so the file entry gets appended.

Of course, none of this behavior is contractual. NTFS would be completely within its rights to, for example, return entries in reverse alphabetical order on odd-numbered days. Therefore, you shouldn't write a program that relies on any particular order of enumeration. (Or even that the order of enumeration is consistent between two runs!)

Raymond Chen

Follow